Scandinavian Journal of Information Systems

Volume 31 | Issue 2 Article 2

12-31-2019

Evaluation of Action Design Research

Stefan Cronholm
University of Borås, stefan.cronholm@hb.se

Hannes Göbel
University of Borås, hannes.gobel@hb.se

Evaluation of Action Design Research

Stefan Cronholm, Hannes Göbel
Department of Information Technology, University of Borås, Sweden
stefan.cronholm@hb.se, hannes.gobel@hb.se

Abstract. Design science research (DSR) is a legitimate research paradigm in the discipline of information systems (IS). One prominent DSR method is Action Design Research (ADR). The purpose of this paper is to evaluate the ADR method based on empirical experiences from a research project. We have found that the ADR method is highly relevant to an applied discipline such as IS. It creates a bridge between the organisational perspective and the technical perspective of the IT artefact. Moreover, the ADR method supports the dual mission of developing theory and producing knowledge that supports IS practitioners. The findings also include empirical evidence pointing towards a lack of prescriptive guidance with respect to the challenges such as: how to identify appropriate evaluation strategies, how to identify the abstraction mechanisms required to move from the specific-and-unique to the generic-and-abstract, and how to formulate design principles. Although we found adequate support at the macro level, the ADR method needs more detailed support for operationalisation in practice. To address this issue, we propose a number of guidelines that either seek to improve the ADR method or support those who apply ADR.

Key words: action design research, ADR, design science research, DSR, ADR evaluation.

1 Introduction

Design science research (DSR) is widely used and regarded as an accepted research paradigm in the discipline of information systems (IS) (e.g., Iivari 2007; Hevner 2007; Gregor and Hevner 2013). The increasing popularity of DSR has created a need for appropriate methods for design-oriented research projects. Judging from the number of citations, Action Design Research (ADR) (Sein et al. 2011) is one of the most popular research methods. Sein et al. (2011) state that ADR is a research method that draws on DSR and action research (AR). The motivation for proposing ADR is that both DSR

Accepting editor: Polyxeni Vassilakopoulou

and AR are insufficient on their own. DSR supports abstraction and innovation while relegating authentic intervention as secondary. AR supports intervention and knowledge emergence in authentic settings but pays little attention to innovation and abstraction (ibid.). ADR is regarded as a "research method for generating prescriptive design knowledge through building and evaluating ensemble IT artefacts in an organizational setting" (Sein et al. 2011, p. 40). By using the term ensemble artefact, and referring to Orlikowski and Iacono (2001), the authors refer to material and organisational features that are socially recognised as bundles of hardware and/or software. Sein et al. (2011) state that the method is especially applicable in research situations where the research question is derived from a problem anticipated in practice, including the naturalistic and formative evaluation of ensemble IT artefacts in a specific context, while searching for new design knowledge. Moreover, Sein et al. (2011, p.37) justify the need for ADR by criticising existing DSR literature in the form of three challenges: 1) "… dominant DR thinking takes a technological view of the IT artifact, paying scant attention to its shaping by the organizational context", 2) "… existing DR methods focus on building the artifact and relegate evaluation to a subsequent and separate phase", and arguing 3) "… a solution to this problem [the relevance challenge] requires a DR method that simultaneously aims at building innovative IT artefacts in an organizational context and learning from the intervention while addressing a problematic situation" (Sein et al. 2011, p.38).

While ADR is being applied more widely, it has not been formally evaluated (see section 2). Our analysis of ADR reveals that its methodological underpinnings are thoroughly theoretically grounded. Sein et al. (2011) present an illustration of how ADR can be applied. The illustration constitutes a reinterpretation of a prior research project conducted at Volvo IT. Sein et al. (2011) state that "…because the VIP [Volvo Information Portal] project was not conducted explicitly as ADR, it cannot be viewed as an exemplar of its application" (p.52) and "The case [Volvo Information Portal] was previously presented in published work as AR" (p.45). We found the reinterpretation of the VIP project valuable as an empirical case illustration. In our literature review of other scholars' use of and reflections on ADR, we found publications: a) on the development of various kinds of design principles, b) including interesting but fragmented reflections on ADR, and c) containing proposals for extensions of ADR. We have also found a number of interesting studies that reflected on ADR, but have not systematically evaluated it. The lack of a formal evaluation of ADR constitutes the research problem of this paper. A formal evaluation is important since it provides the research community with empirical evidence.

Conboy (2009, p.329) state that it is important to distinguish between the 'documented method' and the 'method-in-action'. The documented method is intended to serve as an ideal model for the development process (Iivari and Maansaari (1998), while the method-in-action describes the way it may be used in a particular situation (Fitzgerald et al. 2002). Although Conboy (2009), Iivari and Maansaari (1998) and Fitzgerald (2002) are discussing systems development methods, their ideas concerning the distinction between the documented method and the method-in-action can also inform the evaluation of research methods. Indeed, there are research methods, such as ADR, which are also documented and can be adapted according to specific contextual requirements. Furthermore, systems development can be defined as "… a set of steps (an algorithm or guideline) used to perform a task" (March and Smith 1995, p.257). This definition can also be applied to many research methods. For instance, ADR describes a structure that consists of four stages and associated principles. This means that the aim of this paper is twofold. One aim is to evaluate *ADR in practice (*the method-in-action*)*, which is based on the experiences gained from an ADR project carried out in collaboration with client organisations. This evaluation was summative with the objective of helping those applying ADR. The second aim is to evaluate *the ADR method* (the documented method). This evaluation was formative with the purpose of proposing enhancements to the ADR method. Both the summative and the formative evaluation have resulted in a number of guidelines. Our research question reads: What is the empirical evidence that the ADR method is useful in practice? Haj-Bolouri et al. (2018, p.2) support the importance of the research question when stating that "… there are still uncertainties about how the ADR methodology is being used in practice". Collatto et al. (2017, p.19) add "… it is essential that such proposition be widely applied in practice in order to verify the method's suitability and evaluate its implementation."Furthermore, Collatto et al. (2017, p.19) suggest that "future studies can focus on proposing a more detailed method to operationalize action research design [*sic!*]". By addressing this research question, our contribution concerns prescriptive knowledge including suggestions for applying ADR in practice and proposals for enhancements to the ADR method.

The evaluation, carried out in this study, was part of a larger research project that served several interrelated purposes. One purpose in the overall research project was to design an IT artefact (digital tool) to support service assessment and service innovation in the domain IT Service Management (ITSM). A second purpose was to formulate general design principles concerning the development of IT artefacts supporting service assessment and service innovation. The ADR method was applied to fulfil the first and the second purposes. The ADR method prescribes frequent interaction and the

exchange of knowledge between the development of the artefact and the formulation of design principles. Consequently, the design of the IT artefact and the formulation of the design principles mutually influenced each other. We call this part of the research project *the ADR project* (see section 3). We realized that experiences of applying the ADR method provided an excellent opportunity for evaluating it and its application in practice. Consequently, the third purpose of the research project was to evaluate ADR and this constitutes the purpose of this paper. We refer to this part of the research project as *the ADR evaluation project* (see section 4, 5 and 6).

The research presented in this study extends the authors' prior work (Cronholm and Göbel 2013; Cronholm et al. 2016; Göbel and Cronholm 2016) through: 1) the addition of a thorough state of the art concerning prior evaluations of ADR, 2) the refinement of findings and conclusions based on a re-analysis of data, 3) the presentation of project implications with respect to the ADR principles, and 4) the development of guidelines. This article presents empirical evidence for consideration in future ADR projects. We also argue that the findings could be useful when considering new versions of the ADR method. In the following sections, we describe how our study unfolded through a three-stage process. The first stage included a literature review on applications and evaluations of ADR (see section 2). Its purpose was to describe the state of the art with regard to previous evaluations of ADR, to motivate the need for a systematic evaluation of ADR and to describe how our study advances the state of the art. In the second stage, we describe how the ADR method was applied to design an IT artefact and to develop general design principles (see section 3). In the third stage, we evaluated the experiences of applying the ADR method and developed guidelines (see section 4, 5 and 6). Finally, conclusions are drawn with regard to the evaluation of the ADR method (see section 7).

2 The state of the art concerning evaluation of ADR

In order to examine the state of the art concerning the empirical evaluation of ADR, we reviewed leading IS journals and IS conferences proceedings. Our review included the eight top IS journals ranked by the AIS Senior Scholar's Basket of Journals: European Journal of Information Systems (EJIS), Information Systems Journal (ISJ), Information Systems Research (ISR), Journal of Association of Information Systems (JAIS), Journal of Information Technology (JIT), Journal of Management Information Systems (JMIS), Journal of Strategic Information Systems (JSIS) and, Management Information Systems Quarterly (MISQ). However, in the top eight journals there are only a few studies that report the results of ADR projects and even fewer provided reflections on

ADR. Therefore, we expanded the literature base to include some leading conference proceedings in IS: the Australasian Conference on Information Systems (ACIS), the American Conference on Information Systems (AMCIS), the European Conference on Information Systems (ECIS), the Hawaii International Conference on System Sciences (HICSS), the International Conference on Information Systems (ICIS) and the Pacific Asia Conference on Information Systems (PACIS). The proceedings of the international conference on Design Science Research in Information Systems (DESRIST) were also reviewed. We acknowledge that conference proceedings do not usually hold the same scientific status as journals. However, these articles are of academic standing as they have been peer-reviewed and selected in conference proceedings with a normal acceptance rate of less than 50%. We do not state that our literature review is exhaustive since we limited our search to the AIS basket of eight and leading IS conferences. However, in our opinion the outcome of the literature review sufficiently defined the state of the art with respect to previous empirical evaluations of ADR. This stage also involved limiting the time span of the material collected to the years 2012 and 2018 as ADR was first published in 2011. Table 1 and Table 2 below present an overview of the reviewed journals and conference proceedings, including the numbers of the articles found.

Year	*EJIS*	*ISJ*	*ISR*	*JAIS*	*JIT*	*JMIS*	*JSIS*	*MISQ*	*Total*
2012	1	0	0	0	0	0	0	0	1
2013	0	0	0	0	0	0	0	0	0
2014	1	1	0	1	0	0	0	0	3
2015	1	0	0	1	0	0	1	0	3
2016	0	2	0	1	0	0	0	0	3
2017	5	2	0	0	1	1	0	0	9
2018	6	1	0	0	1	0	0	0	8
Total	14	6	0	3	2	1	1	0	27

Table 1. Number of articles in the reviewed journals

Year	*ACIS*	*AMCIS*	*DES-RIST*	*ECIS*	*HICSS*	*ICIS*	*PACIS*	*Total*
2012	0	2	9	1	1	7	1	21
2013	0	3	8	4	1	6	0	22
2014	5	2	7	6	2	2	2	26
2015	2	3	10	2	1	6	2	26
2016	2	1	6	5	1	8	3	26
2017	1	6	3	3	7	7	0	27
2018	0	7	5	7	11	11	3	44
Total	10	24	48	28	24	47	11	192

Table 2. Number of articles in the reviewed conference proceedings

The literature review demonstrated that ADR has been applied in different ways. Firstly, a majority of the articles dealt with the use of ADR for developing artefacts and suggesting design principles or a design theory of some kind (e.g., Spagnoletti et al. 2015; Ebel et al. 2016; Göbel and Cronholm 2016; Giessmann and Legner 2016; Mettler 2017). These studies used ADR for other purposes than evaluating ADR and are interesting as such. However, they do not provide explicit reflections on ADR, which means that their contribution with respect to the evaluation of ADR is limited.

Secondly, we found studies that provide interesting but fragmented ADR reflections. These studies also had other purposes than the evaluation of ADR. The aim of one such study, conducted by Gregor et al. (2014), was to build capacity for e-government in Bangladesh. An ADR reflection in this study reads: "The project shows, however, how the ADR approach can be used with a real-life problem that encompasses multiple organizations at the level of an entire country" (p. 666). Another study that provides some reflections on ADR was conducted by Mustafa and Sjöström (2013). In their study, the authors argue that it is possible to generalise design principles outside a single case. The purpose of the study conducted by Niemi and Laine (2016) was to deepen

the theoretical understanding of competence management systems. Niemi and Laine (2016) state that ADR supports generalisation by articulating a class of problems and a class of solutions. Schuppan and Koehl (2017) conducted a study about the utility of ADR in e-government, concluding that ADR strengthens a socio-technical view and that it supports co-creation and co-production. A recent study concerning distributed collaboration conducted by Cheng et al. (2018, p.338) concluded that "ADR helps establish in-depth understanding of the relationships between artifacts and organizational contexts, the repeated intervention in this study is an application of the research methodology in real business case."

Thirdly, we found studies that propose modifications or extensions of ADR. For example, Mullarkey and Hevner (2015; 2019) discuss challenges regarding how to enter the ADR research stages effectively. This paper presents eADR as an evolution of ADR that combines ADR with the multiple entry-points in the DSR process suggested by Peffers et al. (2007). In the editorial to the journal, Ågerfalk (2019, p.2) insightfully states "To someone that has followed the DSR discourse in our field, it is probably not surprising that such an elaboration could be somewhat controversial." To shed light on possible agreements and disagreements, Ågerfalk invited two of the authors of ADR (Maung Sein and Matti Rossi) to respond to the suggested evolution of ADR. In the response, Sein and Rossi agreed with some elaborations in eADR such as "… unpacking the specific stages of ADR to make them more transparent and accessible and incorporating formalization of learning in every stage …" (Sein and Rossi 2019). However, Sein and Rossi (2019) strongly disagreed about the suggestion of multiple entry points to an ADR project since the spirit of ADR is that Problem Formulation is the only entry point to the cycle. Sein and Rossi (2019, p.21) also state that "… in juxtaposing the Peffers et al. framework of DSR on to ADR, they [Mullarkey and Hevner 2019] are combining two approaches that are epistemologically incommensurate." The main epistemological difference is that Peffers et al. (2007) deductive approach while "… ADR employs principally an inductive epistemology by giving primacy to the guided emergence of the artifact" (Sein and Rossi 2019, p.21). Another modification of ADR is suggested by Haj-Bolouri et al. (2016). They propose Participation Action Design Research (PADRE), which includes the adoption of principles and philosophy from participatory action research and participatory design. Keijzer-Broers et al. (2016) describe how agile and sprint-oriented design approaches could be integrated into ADR. To sum up, the articles that suggest modifications of ADR, Huysmans and De Bruyn (2013) propose a mixed method approach that embraces how to combine behavioural and design research methods in a coherent manner.

Fourthly, a number of interesting articles that encompass theoretical analyses of ADR were found. While these studies do not contain empirical evidence, they do provide statements and arguments based on theory, arguments and deductive logic. One such study, conducted by Iivari (2015), forwards the argument that the scientific discourse of DSR is unclear and that it is possible to distinguish two DSR strategies. The purpose of the first strategy is to create an IT meta-artefact as a general solution which can be instantiated into a specific solution, such as an IT artefact. The purpose of the second strategy is to "solve a client's specific problem by building a concrete IT artefact (application) in that specific context and distils from it knowledge to be generalized into a general solution concept" (p.107). Iivari (2015) compares the two DSR strategies, with respect to context, outcomes, process and resources. He concludes that ADR relies more on intervention than other DSR approaches. Papas et al. (2012) classify ADR as a meta-approach, which contains elements of both AR and DSR. Furthermore, Papas et al. (2012) state that ADR is driven by a desire to improve the integration of DSR views on design with AR views on evaluation and to move "… away from a limiting software development approach to DS [design science] that fails to capture organisational aspects of the intervention" (p.149). In another theoretical analysis, conducted by Collatto et al. (2017), the authors ask, ' if ADR is indeed necessary?. In response, the authors state that "…the proposals for a new method (action research design [*sic!*]) presented to date are little specific regarding the conduction of research, stressing only macro steps. In addition, studies on action design research are incomplete, lacking a reflection on the circumstances of use or on which research objectives this approach is necessary for and justifiable" (p.19). Furthermore, Collatto et al. (2017) conclude that there is a need for further empirical studies that verify the functionality of ADR. In this respect, we agree with Collatto et al. (2017) and section 1, we motivated the need for a formal evaluation of ADR. To evaluate ADR is important for the IS discipline since it stresses the design of the IT-artefacts to a much larger extent than other IS approaches such as action research.

Finally, we found a few studies with the main purpose of empirically evaluating ADR. For example, the purpose of the study conducted by Rogerson and Scott (2014) was to explore the effectiveness of ADR. Their study of classroom-based training concludes that ADR appears to be an extremely effective research tool. Veling et al. (2016) evaluate how ADR can support exploratory research and design. The authors encountered several challenges that require the development of an alternative ADR variant rooted in an interpretivist paradigm. Haj-Bolouri et al. (2017) present ongoing research and conclude that researchers find it difficult to balance the demands of industry partners and those of the research community. Haj-Bolouri et al. (2018) expanded their pa-

per from 2017 to include an analysis of: how expectations are balanced with the actual outcomes of ADR-projects, how work is coordinated across different ADR-stages, and how to focus on the problem instance is balanced with the class of problems. One conclusion is that ADR is perceived as a high-level framework and that there is a need for more concrete ways of working with the methodology. Their conclusion corresponds well with the purpose of our study.

To summarise, all the studies discussed above provide interesting results and insights. Nevertheless, they either: (a) use ADR to develop design principles or design theories without reflection on or evaluation of ADR, (b) provide fragmented reflections on ADR, (c) suggest extensions to ADR, which implies an explicit or implicit criticism of ADR, (d) analyse ADR from theoretical perspectives, or (e) empirically evaluate ADR from a specific aspect. Unquestionably, the studies related to (a)-(c) above have had different research purposes and were conducted with a research design that did not explicitly support an evaluation of ADR. Studies related to (d) provide interesting theoretical evaluations but lack empirical evaluations. Finally, studies related to (e) consist of empirical evaluations but are limited for different reasons. We can conclude that none of the studies above included a broad and rigorous empirical evaluation of ADR. Consequently, the methodological justifications of ADR stated by Sein et al. (2011) have not been fully empirically evaluated (see section 1). On these grounds, we can conclude that there is a need for a systematic evaluation of the ADR method based on an empirical ADR project.

3 The ADR project

3.1 The dual mission

Sein et al. (2011) recommend that IS projects be organized with respect to the dual mission of making theoretical contributions and assisting in solving the current and anticipated problems of practitioners. In order to meet this recommendation, the ADR project included two closely related purposes. As mentioned in section 1, the first purpose included building an IT artefact (digital tool) that supported practitioners in collaboratively assessing service delivery and innovate services in the domain of ITSM. The practitioners were IT service providers and customers. The first purpose was based on the fact that the practitioners lacked structured support for collaborative service assessment and service innovation. Our problem analysis revealed that service assessment and service innovation were often based on unplanned and unstructured discussions

between the service providers and the customers. The second purpose was to develop general design principles concerning IT artefacts enabling service assessment and service innovation. The fulfilment of the two purposes was not organised as two isolated processes, rather, frequent interaction between different activities in the project was required. In order to fulfil the two purposes of the ADR project, we followed the stages, principles and tasks formulated in the ADR method. The four stages in ADR includes: Problem Formulation; Building, Intervention and Evaluation (BIE); Reflection and Learning; and Formalization of Learning (see Sein et al. 2011 for an exhaustive description).

In order to support the interpretation of the results of the evaluation of the ADR method, this section presents: contextual characteristics concerning the overall research project, which included the ADR project (see section 3.2), information concerning the ADR project with respect to the design of the artefact (see section 3.3) and information about the ADR project with regard to the formulation of the design principles (see section 3.4). We have chosen to describe the project information related to the design of the artefact and the formulation of the design principles separately. In practice, the emergence of the artefact interplayed with the formulation of the design principles. Information concerning conditions, methods and results related to the first and second purpose are presented in detail in Göbel and Cronholm (2016).

3.2 Contextual characteristics

The ADR project comprised four researchers and 15 practitioners from eight organisations including IT service providers and customers. The organisations were facing the same problem, which was a lack of support on how to collaboratively assess service delivery and innovate IT services. In ADR terminology, the eight organisations represented the clients. The ADR project included both private and public organisations of various sizes (see Table 3. Project members). Three organisations had IT services as their core business and five organisations had other core businesses that were strongly dependent on IT services. The practitioners had a high level of pre-knowledge with respect to IT projects, but no pre-knowledge of ADR. The ADR project also included four researchers from one university. The researchers had a high level of pre-knowledge of IT projects and ADR.

The ADR project was characterised by close collaboration and interaction between researchers and practitioners in all the four stages. Another characteristic was that there was a high degree of knowledge exchange between researchers and practitioners, and between the practitioners themselves. The ADR project lasted for three years (2014-2016)

Sector	*Size*	*Roles of practitioner/researcher*
Car Industry	Large	IT Quality Manager, ITSM Process Owner, IT Process Framework Manager
European Clearing House	Large	Acting head of Payment Products and Services, *Executive Vice President Project Management*
IT	Small	Management consultant
IT	Medium	2 Senior Consultants
IT	Medium	Manager Consumer Sales, Service Development Manager
Municipality	Medium	Manager e-services, Customer relationships
Municipality	Small	Manager IT operations, Development strategist
Telecom	Large	Supply Chain Manager
University	Medium	Professor information systems, Associate professor information systems, Assistant professor information systems, PhD student information systems

Table 3. Project members

and was conducted in the domain of ITSM. ITSM can be regarded as an intersection of service science, organisational management and IT. Cronholm and Göbel (2016) define ITSM as a process-based and customer-oriented practice for the management of IT as a service. ITSM is also an umbrella term that includes several best practices and standards, such as ITIL (e.g., Karu et al. 2016), Capability Maturity Model Integrated for Services (Team 2010) and ISO/IEC 20000 IT Service Management Standard (ISO/IEC 2011).

As mentioned above, the context of the ADR project consisted of several organisations. It is debatable whether the ADR method is restricted for use in the context of one single organisation or if it is also applicable in the context consisting of several organi-

sations. This issue is not specifically discussed in Sein et al. (2011). In order to be clear, we asked two of the authors of ADR for guidance. Both authors confirmed that it is possible to use the ADR method in a context that includes more than one organisation and that they could not foresee any obstacles. Consequently, the context in the ADR project spanned over individual organisational boundaries and consisted of a group entity context. A group entity context is defined as: "… a collection of entities, which share certain characteristics, interact with one another or have established certain relations between each other." (Zimmermann et al. 2007). Characteristics that members of a group share may include interests, skills and culture (ibid.). In the ADR project, the organisations: a) shared a common interest in developing a solution for systematically assessing and improving the delivery of IT services, b) shared skills by interacting with each other in order to learn more about a service-oriented perspective in general and about service delivery specifically, and c) shared a common ITSM culture such as process orientation and service thinking. Consequently, these shared characteristics enabled collaboration in the ADR project in meeting the project goals. The decision to form a group of organisations within the ADR project, instead of collaborating with one single organisation, was to increase the possibility of collecting a richer variety of aspects concerning service assessment and the innovation of IT services. The fact that several organisations participated strengthened the formulation of both the problem and the solution.

One important principle in the ADR method that is closely related to the organisational context is Guided Emergence. The idea behind this principle is that "… the artifact emerges from interaction with the organizational context …" (Sein et al. 2011). In the ADR project, the IT artefact emerged from frequent interventions in several organisations that shaped and reshaped the IT artefact and the design principles. During the interventions, anticipated as well as unanticipated consequences were identified and analysed throughout the process. In this way, the emergent nature of the IT artefact was captured. However, the IT artefact did not only emerge from dyadic interventions that included one researcher and one organisation. The fact that several organisations were included meant that we had to organise an arena that included all the participating organisations. The purpose of the arena was to create a meeting place where all of them could interact and learn from each other. In this arena, design related issues identified in the dyadic interventions were discussed, enhanced and agreed upon. Consequently, the IT artefact also emerged from a context that consisted of several organisations sharing the same characteristics.

3.3 Support for assessing service delivery and service innovation

In the ADR stage Problem Formulation, the problem discovered in practice was framed and the theoretical bases were identified. As mentioned above, the problem consisted of a lack of structured support for collaborative service assessment and service innovation. The formulation of the problem was based on interviews with practitioners and workshops that included all the organisations. The problem formulation was also inspired by theories and perspectives concerning open innovation (e.g., Chesbrough et al. 2011) and service-dominant logic (e.g., Vargo and Lusch 2008a; Lusch and Nambisan 2015). The stage Problem Formulation also included identifying and jointly agreeing on the goals of the IT artefact. The most important goals were: a) to facilitate feasible and viable service assessment and service innovation, b) to support co-creation between service providers and customers, c) be easy to learn and use, and d) embed a modern "service innovation and value co-creation" culture. The arguments for the goals were identified in the literature (c.f. Vargo and Lusch 2004; 2008) and through interviews with practitioners.

In the second ADR stage Building, Intervention and Evaluation, the problem formulation and the selected theories were used to formulate the initial design of the IT artefact. In total, the ADR project consisted of three cycles. These corresponded well to the generic schema for the organization-dominant BIE (see Sein et al. 2011). Consequently, the primary source of innovation in the ADR project was the environments of the participating organisations where we challenged existing organisational routines by designing and implementing a novel IT artefact. The argument for favouring the organization-dominant BIE approach was that the research problem was derived from the organisations' contexts. This fact did not mean that technical issues concerning the IT artefact were unimportant or ignored in the ADR project. During each BIE cycle, we analysed and re-analysed the problem, the IT artefact, and the context.

The design of the IT artefact was based on requirements specified by the organisations and on theoretical insights. The actual coding of the IT artefact was carried out by a systems developer employed by the university. The systems developer contributed technical insights and know-how from recent technologies and architecture. Knowledge about the IT artefact emerged from intervention in the organisations' environments. We called each intervention an 'evaluation episode'. The term evaluation episode is borrowed from Venable et al. (2016) and is defined as a particular evaluation. The character of the evaluation process was naturalistic, which is always empirical and encompasses all the complexities of human practice in real organisations (Pries-Heje et al. 2008). The selection of naturalistic evaluation also corresponds well with the

organisational BIE of ADR. In total, the three cycles included 25 evaluation episodes, which meant that several evaluation episodes were carried out with each organisation. Each evaluation episode lasted for approximately two hours. A typical evaluation episode included 1-2 service providers, 1-2 customers and 1-2 researchers. Consequently, the emergent design of the IT artefact was heavily based on intervention including the collection of contextual requirements from the participating organisations. In this way, the IT artefact was mutually shaped by the emerging design principles and the organisational characteristics.

The design of the IT artefact consisted of the following steps: (a) the service provider individually assessed different aspects of the service without involvement of the customer, (b) the customer individually assessed different aspects of the service without involvement of the service provider, (c) the service provider and the customer collaboratively analysed the individual assessments, and (d) the service provider and the customer collaboratively suggested improved services. The steps (a) and (b) were conducted in parallel.

Each evaluation episode ended with a discussion between service providers, customers and researchers in the ways in which the IT artefact supported service assessment and service innovation (i.e., if the IT artefact fulfilled its purpose and if it worked in practice). The discussion lasted for approximately one hour. Moreover, after each discussion individual interviews were conducted with the service providers and the customers. The purpose of the discussion and the interviews was to collect and formulate new requirements in order to continue developing the IT artefact. In this way, the design of the IT artefact emerged through a close relationship between the activities of building, intervention and evaluation. In order to safeguard the collaborative aspect, all practitioners agreed on changes before they were implemented in a new version of the IT artefact. The evaluation episodes in the BIE cycles ended when all organisations agreed that the goals of the IT artefact were fulfilled. One conclusion from the final cycle was that the IT artefact supported structured and collaborative service assessment and service innovation. We can also conclude that the relationship between the service providers and the customers was strengthened.

Figure 1 includes one screenshot of the IT artefact that illustrated the Incident Management (IM) process, which is one of the core processes in ITSM. The traditional purpose of an IM process is to support a service provider in restoring normal service operation as quickly as possible and to minimize disturbance for the customer. That is, the conventional scope of an IM process was to support a single service provider, while the design of the IT artefact in our ADR project extended the IM process to include the customer perspective. Consequently, the IT artefact supported the service

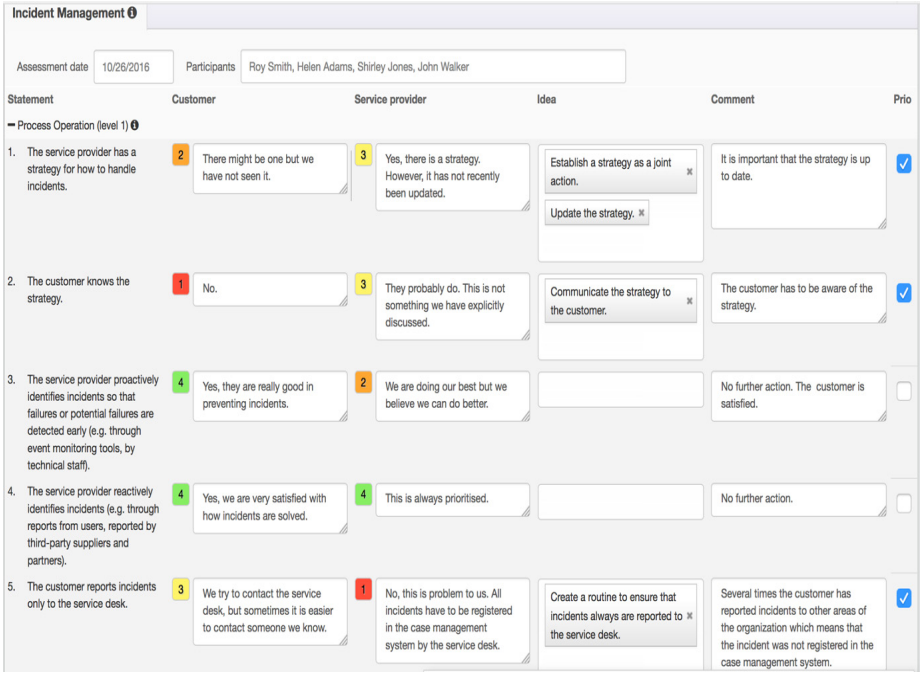

Figure 1. Collaborative service assessment between service providers and customers

providers and the customers to jointly assess the IM process based on predefined statements. As mentioned above, first, each statement was assessed individually by the service providers and the customers. This was done by grading (1-5) each statement and by providing comments to the grading. In a second assessment, the individual assessments were jointly discussed in order to find solutions for statements that were considered problematic. Finally, ideas with respect to how to solve identified problems were documented.

In parallel with the stages Problem Formulation and Building, Intervention and Evaluation, we paid careful attention to the stage Reflection and Learning. In this stage, we analysed instances of problem formulations collected from the interventions in the organisations' environments together with the selected theories in order to formulate a broader class of problems. As mentioned above, the class of problems constituted a lack of structured support for collaborative service assessment and service innovation. The formulation of the class of problems was successively refined due to increased learning about the problem, which in turn affected the design of the IT artefact. Moreover, we

reflected upon the selected theory, the designed IT artefact, the context and the design principles in an integrated fashion.

3.4 Formulation of design principles

As mentioned above, one purpose of the ADR project was to develop generic design principles concerning IT artefacts in order to enable service assessment and service innovation. The fourth stage of the ADR method is called Formalisation of Learning. The objective of the fourth stage was to formalise learning into general solution concepts in terms of design principles. Sein et al. (2011) call this a conceptual move, indicating that there should be a move from the specific-and-unique to the generic-and-abstract. It is important to be aware of the interplay between the general problem formulation and the general solution formulation. In order to clarify the interplay Sein et al. (2011) refers to what DeGrace and Stahl (1990) describe as solving wicked problems. Sein et al. (2011, p.43) state: "For example, the ADR team may use its chosen design constructs to shape its interpretation of the organizational environment, use this increasing understanding of the organizational environment to influence the selection of design constructs, and/or interleave the two." In the ADR project, the design principles emerged as a result of reflection and analysis of the IT artefact in context. In other words, the formulation of the design principles was based on observation of interactions between the service providers and the customers including their interaction with the IT artefact. Their formulations were also based on insights derived from the selected theory. The design principles also governed the design of the IT artefact. In this way, the design principles and the IT artefact mutually influenced each other. Moreover, the design principles were formulated for the class of systems, which we call Innovation Management Systems. Nascent design principles are reported in Göbel and Cronholm (2016) and an enhanced version will be presented in a forthcoming PhD thesis. The formulation of the design principles followed the recommendation by van den Akker (1999). In comparison to other proposals for formulating design principles (e.g., Walls et al. 1992 and van Aken 2004), we have found the proposal by van den Akker (1999) more structured and informative. The design principles are summarised as follows:

Design Principle 1: If you want to design an IT artefact for the purpose of enabling service innovation and service assessment in the context of ITSM, then you are best advised to characterise the intervention as *value co-creation*, and to do that via the procedure *design dynamic change processes uniting all actors in the service ecosystem* on the basis that *access to shared resources will leverage the innovation process*.

Design principle 2: If you want to design an IT artefact for the purpose of enabling service innovation and service assessment in the context of ITSM, then you are best advised to characterise the intervention as co-problematization, and to do that via the procedure design for service provider and customer collaboration concerning problem identification and problem analysis on the basis that it is more likely to solve the real customer's problem.

Design principle 3: If you want to design an IT artefact for the purpose of enabling service innovation and service assessment in the context of ITSM, then you are best advised to characterise the intervention as continuity, and to do that via the procedure design a routine that ensures continual co-assessments and co-innovation on the basis that continual improvements will enable the IT service to up-date with respect to changing customer needs and other environmental changes.

4 Analysis strategy and process

We decided to follow the evaluation framework suggested by Pries-Heje et al. (2008) and Venable et al. (2016). The reason was that the authors have developed a specific framework for the evaluation of design science research (FEDS). The overall purpose of the framework is to support the formulation of an answer to the question: "What would be a good way to guide the design of an appropriate strategy for conducting the various evaluation activities needed?" (p.80). The overall question is further divided into the following two dimensions: 1) the functional purpose of the evaluation (why evaluate) and 2) the paradigm of the evaluation study (how to evaluate). In our study, the functional purpose concerned both formative and summative evaluation. William and Black (1996) state that formative evaluations focus on consequences and support the kinds of decisions that seek to improve the evaluand, while summative evaluations focus on meanings and support the kinds of decisions that seek to influence the selection of the evaluand for an application. Moreover, William and Black (1996) state that an evaluation process that may have been formulated for summative purposes may also be put to use for formative purposes. In our case, the purpose of the formative evaluation was to suggest enhancements to the ADR method, and the purpose of the summative evaluation was to help those using ADR. Venable et al. (2016) state that the paradigm of the evaluation study includes a distinction between artificial evaluation (e.g., laboratory experiments, testing design hypotheses) and naturalistic evaluation (e.g., exploration of the performance of a solution technology in its real environment, typically within an organisation). In our study, the evaluation strategy was naturalistic since the purpose was to explore the performance of the ADR method in real organisa-

tional environments. In other words, the evaluation involved real users, real problems and real systems, which constitute three key ingredients in empirical evaluation (Sun and Kantor 2006).

The evaluation strategy was complemented with qualitative content analysis, as it provides a method for analysing text data (e.g., Hsieh and Shannon 2005). One approach within qualitative content analysis is direct content analysis. Hsieh and Shannon (2005) recommend researchers to use this approach when theory or research already exist about a phenomenon but would benefit from further description. The goal of direct content analysis is to validate a theoretical framework (or a method) and its main strength is that it can support and extend existing theory (ibid.). Mayring (2000) calls this approach deductive category application and argues that prior work (such as a method) can contribute to determining the codes that will be used. In order to apply the direct content analysis approach, our study followed the research process proposed by Seuring and Müller (2008): category/code selection, material collection and material evaluation. As an additional step, we developed guidelines that either propose enhancements to the ADR method or provide guidance on how to apply ADR.

4.1 Step 1: Category/code selection

According to Seuring and Müller (2008), the analytic categories in a deductive approach, such as direct content analysis are selected before the analysis is conducted. The codes we derived consisted of the seven ADR principles: Practice-Inspired Research, Theory-Ingrained Artifact, Reciprocal Shaping, Mutually Influential Roles, Authentic and Concurrent Evaluation, Guided Emergence, and Generalized Outcomes (see Sein et al. 2011 for a detailed description). The ADR principles were chosen as codes because they encapsulate the underlying beliefs and values of the method (Sein et al. 2011). In order to make the evaluation less abstract, we also identified specific statements in the ADR method that further concretised the principles. In this paper, a statement was defined as a prescription concerning what to do or how to do something. The selected statements only included those made by the authors of the ADR method. The statements were identified by a careful reading of the ADR method. First, two of the researchers (the authors of this paper) individually identified the statements. Then, they jointly compared the individually identified statements in order to create an agreed list of statements. Finally, the statements were grouped according to the ADR principles.

4.2 Step 2: Material collection

Our second step was to use the ADR principles and the statements as a lens for collecting a wide range of project implications from the use of the ADR method in the ADR project. As mentioned in section 3, we conducted three cycles that included 25 evaluation episodes in the participating organisations' real contexts. The project implications were identified during these evaluation episodes. According to Mayring (2000), the subject matter of qualitative content analysis can be all kinds of recorded communication, such as transcripts of interviews, discourses, protocols of observations, videotapes, and documents. This meant that we collected the project implications by: 1) Using videotapes; 2) Taking notes on specific comments with respect to the ADR method. Notes were taken from both practitioners and researchers during project meetings and workshops; 3) Gathering experiences from prescribed ADR actions. This meant that we analysed ADR principles and statements: a) in relation to the attributes of the designed IT artefact, b) with respect to the formulation of the design principles, and c) with regard to the benefit of created documents that supported the development process; 4) Interviewing the four researchers who participated in the project in order to collect their individual experiences with respect to their actions carried out concerning the building of the IT artefact, the interventions, the evaluation episodes and the formulation of the design principles.

4.3 Step 3: Material evaluation

Individual evaluation of project implications: Goldkuhl (1999) states that a prescribed action should be evaluated and justified with reference to the actual performance of the action. Similarly, Eisenhardt and Graebner (2007) suggest that explicit links between propositions and empirical evidence should be created. This "… leads to very clear pattern recognition of the central constructs, relationships, and logic of the focal phenomenon" (ibid. p.27). Our analysis followed these recommendations and we created an explicit link between one ADR statement, one or several project implications, and the evaluation of the ADR statement.

In the first evaluation activity, the researchers individually matched the ADR statements identified in step 1 above with the collected project implications in step 2. In this second evaluation activity, the researchers individually evaluated if the ADR statement was easy to follow, or if we had to find support for ways of proceeding elsewhere, or if we had to create a solution ourselves. This exercise corresponded to the summative evaluation of ADR in practice. In the third evaluation activity, the researchers individually compared the purpose and the content of the ADR statements with the character of the

project implications. This evaluation activity was guided by the question formulated by Goldkuhl (1999, p.10): "Is the prescribed action really successful in practice?." The third evaluation activity corresponds to the formative evaluation of the ADR method. In this way, we created "pattern-matched" constructions that consisted of an explicit link between the ADR statements, the project implications from the ADR project and the formative evaluation of the ADR statements. We regard a single construction as a piece of empirical evidence of a specific ADR principle.

Reconciliation of the individual analyses. In an interpretative approach, such as content analysis, the analyst makes various decisions about how to comprehend the data (Walsham 1995). According to Seuring and Müller (2008), the risk of misinterpretation can be reduced by involving two or more researchers when searching and analysing the data. Consequently, the results of the individual evaluations of the ADR statements in relation to the project implications were compared and reconciled in a second analysis that included two of the researchers. The reconciliation of the individual analyses was organised on three occasions, each of which lasted for two hours. The process of reconciliation followed a consensus process, which meant that it was a co-operative process that led to an agreement supported by both researchers (e.g., De-Groth 1974). First, the researchers presented the individual evaluations to each other, then similar and different interpretations of ADR statements in relation to the project implications were discussed (this sometimes required re-analyses of the collected material), and finally the individual evaluations were merged and refined in consensus. The result from the reconciled analysis consists of refined constructions of linked ADR statements, project implications and evaluations of ADR statements. The presentation of the constructions was guided by Eisenhardt and Graebner (2007). They state that a "… a separate table that summarizes the evidence for each theoretical construct is a particularly effective way to present the case evidence." Eisenhardt and Graebner (2007, p.29) motivate their statement by concluding: "These "construct tables" summarize the case evidence and indicate how the focal construct is "measured," thus increasing the "testability" of the theory and creating a particularly strong bridge from the qualitative evidence to theory-testing research." We have followed this recommendation and the constructions are presented in a number of tables that are related to the ADR principles (see section 5).

4.4 Step 4: Development of guidelines

Based on the summative and formative evaluations, we suggested a number of guidelines. The purpose of the guidelines is either to provide guidance on how to apply ADR

in practice or to suggest enhancements to the ADR method. The guidelines concerning the application of ADR in practice emerged from the project implications. The guidelines regarding suggestions for enhancements to the ADR method emerged from the evaluation of the ADR statements and the ADR principles in relation to the character of the project implications. Moreover, the guidelines were evaluated by implementation and use in the ADR project and have been successively refined during the evaluation of the ADR method. In our literature analysis, we identified suggestions for the formulation of design principles regarding the development of IT artefacts (e.g., Walls et al. 1992; van den Akker 1999; van Aken 2004; Chandra et al. 2016; Cronholm and Göbel 2018). These suggestions inspired us to formulate formal guidelines on how to apply ADR in practice as well as enabling us to suggest enhancements to the ADR method. We decided to follow the suggestion by Walls et al. (1992) as it is formulated on a general level, making it more widely applicable. The formulation by Walls et al. (1992, p.41) reads: "If you want to achieve goal X, then make Y happen" (see section 6).

5 Findings

The evaluation of ADR was structured according to the seven ADR principles: Practice-Inspired Research (Problem Formulation), Theory-Ingrained Artifact (Problem Formulation), Reciprocal Shaping (Building, Intervention and Evaluation), Mutually Influential Roles (Building, Intervention and Evaluation), Authentic and Concurrent Evaluation Building, Intervention and Evaluation), Guided Emergence (Reflection and Learning), and Generalized Outcomes (Formalization of Learning). Each subsection includes a short description of the principle and a description of the findings. In order to provide transparency, we have provided quotes from both practitioners and researchers involved in the ADR project and examples of screenshots from the designed IT artefact. We have structured the findings in a table consisting of three columns. The first column contains ADR statements that are related to the ADR principle. The ADR statements include normative prescriptions, recommended actions or objectives that should be fulfilled. The second column includes the implications of the ADR project, with respect to the ADR statements. This column corresponds to the summative evaluation of ADR in practice. The third column contains the evaluation of the ADR statements, which is based on a comparison of the purpose and content of the ADR statement and the character of the project implication. This column corresponds to the formative evaluation of the ADR method.

5.1 Principle 1: Practice-inspired research

"This principle emphasizes viewing field problems (as opposed to theoretical puzzles) as knowledge-creation opportunities." (Sein et al. 2011, p. 40)

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"… the action design researcher should generate knowledge that can be applied to the class of problems that the specific problem exemplifies" (Sein et al. 2011, p. 40)	The development of a class of problems encouraged the researchers to make abstractions, in order to gain deeper knowledge both about the class of the problem and instances of the problems. More specifically, the class of problems was identified by moving from the specific-and-unique (i.e., individual interviews and evaluation episodes with the organisations) to the generic-and-abstract (i.e., workshops including all the organisations and researchers, insights from theory). In this way, both theoretical and empirical arguments were considered. The generic-and-abstract formulations were also supported by conducting a root-cause analysis (Wilson et al. 1993). First, we carried out individual analyses with the eight organisations. Then, the results were consolidated into one generic root-cause diagram, which was jointly discussed and accepted by all the organisations. The root problem consisted of a lack of design principles on designing IT artefacts that support the collaborative assessment of service delivery and service innovation. Moreover, there was also a lack of instantiated IT artefacts for solving the problem. The root problem was identified as due to: a) problems perceived in practices such as the fact that ITSM best practices are not based on contemporary service perspectives, b) lack of knowledge regarding normative and prescriptive guidelines supporting service-orientation.	This ADR statement emphasises the importance of deeper learning and distinguishes ADR as a research method from a systems development method.

"The intent of the ADR team should not be to solve the problem per se as a software engineer or a consultant might" (Sein et al. 2011, p. 40)	The statement strengthened the researchers' confidence in acting as researchers. It supported us in formulating and explicitly communicating the research question to the organisations. Another example is that the development of questionnaires reflected both questions of interest to the practice and identified gaps in the theory. Examples of questions asked were: what are the core ITSM processes in your organization? And, how do you co-create value with the service customer?	The ADR statement legitimates and affirms the research interest without neglecting the goals of the practitioners.
"Cast the problem as an instance of a class of problems" (Sein et al. 2011, p. 41)	There was a lack of prescriptive support in the ADR method for how to cast the problem as an instance of a class of problems. Therefore, we were inspired by theories concerning object-orientation, which particularly supports: a) the formulation of instances and classes, and b) the identification of their relationships. The bottom-up approach suggested in ADR with respect to the formulation of a class of problems was complemented with a top-down approach. The formulation of the class of problems reads: a lack of structured support for collaborative service assessment and service innovation.	The concepts and processes of working with instances and classes were not explained in enough detail.

Table 4. Evaluation of principle 1: Practice-inspired research members

One purpose of principle 1 is to move conceptually from building a solution for a particular instance of a problem to applying a solution to a broader class of problems (see Table 4). Striving for broader learning and general solutions is widely recognised in DSR (e.g., March and Smith 1995; Hevner et al. 2004). Our analysis has revealed positive experiences concerning the creation of a class of problems since the recommendation promotes learning and legitimises research as a profession. It also advises against a type of IT consultant behaviour that consists of solving an instance of a problem. A quote from one of the researchers is: "To identify a class of problems increases the possibility that the problem is true, new and interesting." However, one of the project implications regards the lack of prescriptive guidance for the creating of a class of problems. The use of the concepts of class and instance indicates that Sein et al. (2011)

are inspired by the object-oriented theory (e.g., Date 2006). ADR encourages users to identify the properties of an instance that could also be valid for the class. The use of a bottom-up approach is a good way to create a class. However, the creation of classes of instances also requires a top-down approach, since it is necessary to test a new instance in the class against other instances to verify that it belongs to the class (ibid.). Consequently, the concepts and process of working with instances and classes could be explained in more detail.

5.2 Principle 2: Theory-ingrained artefact

"This principle emphasizes that the ensemble artifacts created and evaluated via ADR are informed by theories." (Sein et al. 2011, p. 40)

A cornerstone of ADR is the inclusion of the organisational context in the design of the IT artefact. In the project, the principle of theory-ingrained artefact was considered positive since it created a good balance between the inscription of organisational and theoretical knowledge into the IT artefact (see Table 5). One quote from one of the practitioners is: "We are happy to include ideas from researchers. In traditional systems development projects we would never have considered merging requirements derived from theory, such as the foundational premises of Service Dominant Logic, with our own requirements." During the ADR project, this principle encouraged us to be much more explicit regarding the theoretical elements' influence on the IT artefact. Another quote from one of the researchers is "This principle supported the integration of theory into the IT artefact. It also helped us to explicitly trace theoretical inscriptions in the IT artefact, which was used later as arguments for the successful application of theory." The principle also supported us not only in our use of existing knowledge but also in a cumulative way in further developing existing knowledge through the integration of new empirical knowledge.

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"… the action design researcher actively inscribes theoretical elements in the ensemble artefact …" (Sein et al. 2011, p. 41)	Inscriptions of theoretical elements were made explicit and part of the requirement specification. Service-oriented theoretical elements inscribed in the IT artefact were inspired by the fundamental premises (FPs) concerning value co-creation (Vargo and Lusch 2008). Figure 1 (see section 3.3) illustrates some examples of how the concept of value co-creation guided the design of the artefact. The process used for illustration is Incident Management (see section 3.3). The FP "A service-centered view is customer oriented and relational" guided the selection and formulation of the statements that were used for assessment. The FPs "The application of specialized skill(s) and knowledge is the fundamental unit of exchange" and "The customer is always a co-creator of value" highly influenced the design of a collaborative situation including the service providers and the customers in assessing the statements. The collaborative assessment process also included suggesting innovative ideas that could solve the problems revealed by the assessment. Another project implication was that this ADR statement encouraged the researchers to include explicit illustrations of how the theory was inscribed into the IT artefact in research papers.	The inscription of theory into the IT artefact reduced the gap between theory and practice. The recommendation to inscribe theory into the IT artefact supported service providers and customers to share knowledge and skills. The shared problem understanding provided good conditions for co-creating valuable services.

Table 5. Evaluation of principle 2: Theory-ingrained artefact

5.3 Principle 3: Reciprocal shaping

"This principle emphasizes the inseparable influences mutually exerted by the two domains: the IT artifact and the organizational context." (Sein et al. 2011, p. 40)

An overall experience related to principle 3 is that ADR is useful for research problems that require frequent interventions in organisations, in order to establish an in-depth understanding of the artefact-context relationship (see Table 6). Sein et al. (2011) specifically advise researchers to intervene in authentic settings. Unquestionably, the organisational context is emphasised in ADR, especially in the BIE stage. However, there is no specific guidance regarding how to identify, manage and finally inscribe contextual characteristics into the design of the IT artefact. One comment from several practitioners regarded a lack of understanding of the concept of context. They specifically asked "What is context?" and "How can we identify contextual characteristics that affect the design of the IT artefact?" The latter comment was also valid for the researchers. In the ADR method, there is a lack of guidance with respect to the identification of contextual characteristics. We found this crucial since the objective of evaluation is the ongoing refinement of the IT artefact as it is shaped and reshaped by its context of use (Sein et al. 2011).

5.4 Principle 4: Mutually influential roles

"This principle points to the importance of mutual learning among the different project participants." (Sein et al. 2011, p. 43)

ADR maintains that researchers and practitioners should agree upon their roles and responsibilities and establish a formal researcher-client agreement. We deemed our experience of these recommendations concerning collaboration to be important, as they clarify the interests of both the researchers and the practitioners (see Table 7). One of the practitioners stated that "If we look at how practitioners in general behave, no one is searching for theoretical support when they are trying to find a solution to a problem. Thus, this principle is brilliant and should be incorporated into the methods we are using."

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"ADR is useful for … intervention in organizations to establish the in-depth understanding of the artefact-context relationship" (Sein et al. 2011, pp. 52-53)	The shaping of the IT artefact was improved by frequent intervention in the organisational context. This interaction increased organisational knowledge, which improved the quality of the IT artefact. For example, we implemented assessment statements concerning service delivery in the IT artefact such as: "The service provider proactively prevents incidents" and "The customer is always accessible to contribute with the necessary information for solving the incident." Such assessment statements, as well as technical functionality, were iteratively modified according to the contextual requirements identified during the interventions.	The ADR statement supported intervention and an in-depth understanding of the relationship between the artefact and the organisational context.
"The goal of this large-scale evaluation is the ongoing refinement of the artefact as it is shaped and reshaped by the use context" (Sein et al. 2011, p. 42)	There is no explicit support in the ADR method for how to identify contextual requirements that should be inscribed into the IT artefact. In the ADR project, we applied process analysis (vom Brocke and Rosemann 2010) and root-cause analysis (Wilson et al. 1993). We used these tools to successively collect requirements in order to shape and reshape the IT artefact during the cycles in the ADR project. Examples of contextual requirements that were implemented in the IT artefact were: tight collaboration between service providers and customers, resource focus, and process-, and service-orientation.	There is a lack of guidance regarding how to identify and inscribe organisational aspects into the IT artefact.

Table 6. Evaluation of principle 3: Reciprocal shaping

One aim of the researcher-client agreement is to engage the practitioners as active co-creators of knowledge instead of as passive information providers. A quote from one of the researchers is: "This principle and the researcher-client agreement guided us to organise the ADR project in a way that encouraged learning between practitioners and researchers, and between practitioners themselves." We also experienced that some of the participating organisations, due to a lack of time and sometimes motivation, were primarily interested in a solution (instance) that solved their own specific problems and were not primarily interested in finding a class of solutions. In the ADR project, this observation constituted a potential conflict between the interests of the researchers and those of the practitioners.

We found it to be a weakness that ADR is developed by researchers for researchers. In true collaborative researcher-practitioner context practitioners and researchers interact and mutually influence each other. We found that practitioners sometimes experienced the language in ADR as too abstract. Examples of such abstract concepts include construct, ensemble artefact and theory-ingrained artefact. Of course, such concepts are not common in the vocabulary of practitioners. The problem of communicating academic constructs to practitioners is a general problem and does not only apply to ADR projects. To maintain the collaborative aspect of ADR projects, researchers need to find suitable translations.

5.5 Principle 5: Authentic and concurrent evaluation

"This principle emphasizes a key characteristic of ADR: evaluation is not a separate stage of the research process that follows building." (Sein et al. 2011, p. 43)

ADR advocates close interplay between design and evaluation. Sein et al. (2011) emphasise that decisions about the design of the artefact and intervention in organisational work practices should be interwoven with ongoing evaluation. Peffers et al. (2018, p.135) add that "While most DSR methodologies view design and evaluation as sequential, ADR envisions design and evaluation as one process that emerges in researcher/organization interaction." In the ADR project, the inseparability of design and evaluation was considered a strength, since it supported the inscription of contextual factors throughout the process of building, intervention and evaluation (see Table 8). One comment from one of the practitioners in the ADR project supported the importance of a close relationship between the design and evaluation by stating: "this approach differs from the agile methods we use in our organization, since there is so much emphasis on evaluation and reflection."The researchers agreed with this statement and added

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"… set up the roles and responsibilities" (Sein et al. 2011, p. 40)	Establishing roles and responsibilities in advance reduced possible misunderstandings and ensured the assignment of the roles (competence) needed to solve the problem. In the ADR project, a letter of intent was initially created as a mutual agreement between the researchers and practitioners.	The emphasis on formal agreements concerning roles and responsibilities clarifies the parties' expectations of each other.
"… the situated learning from an ADR project should be further developed into general solution …" (Sein et al. 2011, p.44)	To support broader and deeper learning in order to develop a general solution, we invited all the organisations to a workshop. The purpose of the workshop was to share knowledge and to identify a general solution. A majority of the participating organisations were highly motivated since they learned from each other. In this way, the general solution was based on synergy effects from mutual learning in the organisations. However, we also experienced that some organisations were primarily interested in a solution that addressed their own specific business problem and less interested in finding a general solution. To satisfy these organisations' requirements we implemented flexible ways of using the IT artefact.	There is no prescriptive guidance that advises how to engage organisations in the development of a general solution.
"ensemble artefact" (Sein et al. 2011, p. 38) "theory-ingrained artifact" (Sein et al. 2011, p. 40) "design constructs", (Sein et al. 2011, p. 43)	Some terms and concepts in ADR were considered too abstract. Abstract ADR concepts were translated to a language that was used in the practitioners' organisations. Consequently, we often referred to the ensemble artefact as the IT system or the web application. We used the term 'requirements identified in theory' instead of theory-ingrained artefact and we used the word 'concepts' instead of design constructs.	Some concepts used in the ADR method constitute communication barriers between researchers and practitioners.

Table 7. Evaluation of principle 4: Mutually influential roles

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"… evaluation is not a separate stage of the research process that follows building" (Sein et al. 2011, p. 43)	The close interplay between building and evaluation supported iterative development with more frequent interaction between researchers and practitioners. However, in the ADR method there is no prescriptive support concerning how to evaluate the IT artefact. This meant that we consulted evaluation strategies presented by Pries-Heje et al. (2008) (see section 3.2). In each BIE cycle, we applied evaluation criteria such as: utility, support for collaboration, performance and fit for the context. The evaluation criteria were derived from the goals and they were measured during the evaluation episodes. The evaluation criteria were agreed on by all the organisations.	The principle of authentic and concurrent evaluation together with the principles reciprocal shaping and mutually influential roles supported the inseparability of building and evaluation. There is no prescriptive guidance concerning authentic evaluation strategies.

Table 8. Evaluation of principle 5: Authentic and oncurrent evaluation

that "the integration of design, intervention and evaluation was considered helpful." However, they also stated that "the lack of prescriptive guidance concerning evaluation constituted a barrier that had to be navigated".

5.6 Principle 6: Guided emergence

"It [Principle 6] emphasizes that the ensemble artifact will reflect not only the preliminary design (see Principle 2) created by the researchers but also its ongoing shaping by use, perspectives, and participants." (Sein et al. 2011, p. 44)

Guided emergence stresses that new requirements should be identified and inscribed into the initial design during the BIE iterations thus allowing the emergence of a contextualised IT artefact. Consequently, organisational characteristics were jointly discussed by researchers and practitioners. The characteristics were subsequently transformed into new requirements, inscribed into the IT artefact and finally evaluated in the contexts of the organisations (see Table 9). One quote from one researcher reads: "The combina-

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"Conscious reflection on the problem framing, the theories chosen, and the emerging ensemble is critical to ensure that contributions to knowledge are identified." (Sein et al. 2011, p. 44)	In the ADR project, the interplay between theory, organisational intervention and the emergence of the IT artefact supported reflection and the formulation of design principles. Different versions of the design principles were documented and stored, which made it possible to visualise, communicate and reflect upon the emerging knowledge.	The ADR statement supports a continuous shift of focus between the organisational intervention, the emergence of the IT artefact and the development of knowledge.
"ADR reaches into the very core of IS: designing IT artifacts while allowing for their emergence in an organizational context, and seeking utility in the ensemble they represent." (Sein et al. 2011, p. 53)	The IT artefact emerged through intervention and awareness in social contexts and from environmental influences. One of the main contributions of an ADR project is utility for the users. We measured utility for the users by comparing the situation before and after the implementation of the IT artefact.	This ADR statement supports the understanding that: 1) IT artefacts are not fixed; they emerge from social contexts, and 2) IT artefacts are not static; they are dynamic and change over time.

Table 9. Evaluation of principle 6: Guided emergence

tion of guided and emergence provided a balance between governance and flexibility." Another quote from one of the researchers concerns the lack of concretion with respect to this principle. The researcher stated that "It would be helpful if there were more examples of how guided emergence can be applied in order to shape the artefact."

5.7 Principle 7: Generalized outcomes

"The resulting ensemble is, by definition, a bundle of properties in different domains. This ensemble represents a solution that addresses a problem. Both can be generalized. This move from the specific-and-unique to generic-and-abstract is a critical component of ADR." (Sein et al. 2011, p. 44)

The generalised outcome of an ADR project consists of design principles. Design principles are created in order to capture the knowledge gained about the process of building solutions for a given domain and to incorporate knowledge about creating other instances that belong to this class (Sein et al. 2011). ADR advocates a generalisation process that follows the "move from the specific-and-unique to the generic-and-abstract", which supported our focus on developing generic design principles (see Table 10). However, ADR does not provide prescriptive support regarding generalisation or how to formulate design principles. A quote from one of the researchers supports this observation and poses the question: "What is the anatomy of a design principle?." Another quote that concerns the abstraction of knowledge, phrased by one of the practitioners is: "I do not understand the difference between a design principle and a system requirement."

Sein et al. (2011, p. 44) recognise that "Generalization is challenging because of the highly-situated nature of ADR outcomes …" (p. 44). Haj-Bolouri et al. (2017) also recognise generalisation as challenging due to the contextual nature of design. We recognise the attached Volvo case in ADR as an excellent example and summary of how the ADR principles have been justified, but the case does not provide guidance regarding how to generalise outcomes or formulate design principles. Thus, in order to find support for abstraction mechanisms and generalisation, we consulted Gregor and Hevner (2013). This support was considered helpful since we managed to address two of the three levels described: situated implementation of artefact and nascent design theory (knowledge as operational principles/architecture). The third level which consists of a well-developed design theory about embedded phenomena will be presented in a forthcoming PhD thesis. The process of generalisation was also supported by the fact that several organisations participated in the project. We viewed each organisation's specific need as an instance, which provided a base for generating the class of solutions. In other words, the class of solutions was generalised from several organisations' unique needs and settings.

To find support for the formulation of design principles, we consulted the logic suggested by Van den Akker (1999). This logic proposes that design principles should be described according to the following aspects: purpose, context, characteristics, pro-

ADR statement	*Project implication*	*Evaluation of the ADR statement*
"… reconceptua-lizing the learning from the specific solution instance into design principles for a class of solutions" (Sein et al. 2011, p.45)	The generalised outcome was expressed as design principles for a class of solutions (i.e. design principles for the development of IT artefacts, with respect to service assessment and service innovation). The design principles were conceptualised through reflection and learning from the evaluation episodes. We reflected upon design decisions concerning the functionality of the IT artefact and on the process of design decision-making. Consequently, the shaping and reshaping of the IT artefact influenced the formulation of the design principles.	Knowledge creation is supported by the recommendation to generalise outcomes.
"… move from the specific-and-unique to the generic-and-abstract" (Sein et al. 2011, p.44)	Due to a lack of support regarding how to generalise, we consulted prescriptive theories concerning abstraction mechanisms in terms of generalisation and specialisation (Gregor and Jones 2007; Gregor and Hevner 2013). These theories contributed with an understanding of how IT artefacts, design principles and design theories can be regarded as interrelated levels of abstraction.	Lack of prescriptive guidance concerning generalisation and abstraction.
"Articulate outcomes as design principles." (Sein et al. 2011, p.45)	Due to a lack of support on how to formulate design principles, we followed the guidelines presented by Van den Akker (1999). The guidelines ensured that the design principles were consistent with regard to content and format.	Lack of prescriptive guidance regarding how to formulate design principles.
"Formalize results for dissemination." (Sein et al. 2011, p.45)	In order to disseminate the results of the ADR project, papers were jointly authored by researchers and practitioners. The papers constituted research papers, reports and technical papers (white papers). The ADR project also organised several conferences for both researchers and practitioners. Moreover, the results were implemented in academic courses. All the dissemination activities strengthened the researcher-client relationship.	ADR has a strong focus on the formalisation of results. ADR does not explicitly mention researchers and practitioners as co-authors of scientific papers. However, we found excellent examples (e.g., Westin and Sein 2015; Göbel and Cronholm 2016).

Table 10. Evaluation of principle 7: Generalised outcomes

cedures and arguments. A final project implication concerns the dissemination of the ADR project's results. In order to strengthen the relationship with the practitioners and to improve access to empirical knowledge, a variety of papers were co-produced.

6 Proposal for guidelines

Section 1 presents three challenges formulated by Sein et al. (2011) which constitute a criticism of existing DSR literature. According to Sein et al. (2011), the challenges justify the need for ADR. The purpose of this section is to discuss further how ADR responded to the challenges concerning the results of the evaluations conducted in section 5. The challenges are somewhat overlapping, which means we evaluated them with respect to their main foci: ensemble artefact (challenge 1), inseparable building, intervention and evaluation (challenge 2); and intervention in the organisational context (challenge 3).

Some of the project implications presented in section 5 included the lack of prescriptive knowledge. Goldkuhl (2004) states that for specific goals to be reached, design methods should provide prescriptions for actions. The need for prescriptive actions is also acknowledged by Collatto et al. (2017). They maintain that "… it's necessary to develop researches and publications to propose clear and detailed procedure to conduct the action design research" (p.19). In the cases we have found a lack of prescriptive knowledge, we propose a number of guidelines that either suggest enhancements to the ADR method or provide guidance on how to apply ADR.

6.1 Challenge 1: The ensemble artefact

"[D]ominant DR thinking takes a technological view of the IT artifact, paying scant attention to its shaping by the organizational context." (Sein et al. 2011, p.37).

The response of Sein et al. (2011) to challenge 1 is to encourage the development of ensemble artefacts, which emanate from the term ensemble view of artefacts coined by Orlikowski and Iacono (2001). We found that the development of ensemble IT artefacts is mainly supported by the ADR principles: Reciprocal Shaping, Mutually Influential Roles, and Guided Emergence. These principles helped us since they emphasise the importance of shaping the artefact with regard to the organisational context. In our literature review (see section 2), we paid attention to the epistemological discussion between Mullarkey and Hevner (2019) and Sein and Rossi (2019). One part of the discussion concerned whether it is appropriate to combine ADR with the DSR method

as suggested by Peffers et al. (2007). Sein and Rossi (2019, p.21) state that Mullarkey and Hevner (2019) combine two approaches that are epistemologically incommensurate since "… ADR employs principally an inductive epistemology by giving primacy to the guided emergence of the artifact", while the method suggested by Peffers et al. (2007) is deductive. Based on the findings in our study, we agree with Sein and Rossi (2019) and can conclude that the ADR project to a large extent relied on intervention in the organisational context and that the IT artefact emerged from inductive analyses of empirical data. However, we also appreciated recommendations that supported the development of the IT artefact through theoretical insights.

Sein et al. (2011, p.38) state that "Designing ensemble artifacts involves dimensions beyond the technological, because they result from the interaction of design efforts and contextual factors throughout the design process." We appreciated the normative aspect (*what*) of this statement but did experience a lack of prescriptive guidance, with respect to *how* the contextual factors can be inscribed into the IT artefact for service assessment and innovation. In other words, the emergence of the IT artefact needed guidance since it was not fixed; it emerged from social contexts, which meant that complementary method support had to be found. The strategies suggested by Pries-Heje et al. (2008) and later developed by Venable et al. (2016) helped us in identifying contextual factors and selecting an appropriate evaluation strategy, which was a naturalistic evaluation. Naturalistic evaluation enabled us to explore the performance of the artefact in real environments. In addition, we consulted methods for process analysis (vom Brocke and Rosemann 2010) and root-cause analysis (Wilson et al. 1993). We are not stating that the ADR method should include a section concerning evaluation strategies. However, to enhance the guidance of ADR projects, we suggest that the ADR method should consist of references to relevant evaluation literature. Based on shortcomings concerning the lack of prescriptive guidance, we propose the following guidelines:

Guidelines concerning enhancements to the ADR method
In order to provide support for evaluation strategies, the ADR method should:

- Include prescriptive guidance regarding *how* the contextual factors can be inscribed into the IT artefact.

- Offer references to DSR evaluation literature.

Guidelines concerning how to apply ADR in practice
In order to ensure that contextual characteristics inscribed into the IT artefact:

- Plan for the acquirement of knowledge through naturalistic evaluation (e.g., Pries-Heje 2008; Venable et al. 2016).

- Analyse organisational process (e.g., vom Brocke and Rosemann 2010) and conduct root cause analysis (e.g., Wilson et al. 1993).

6.2 Challenge 2: Inseparable building, intervention and evaluation

"[E]xisting DR methods focus on building the artifact and relegate evaluation to a subsequent and separate phase." (Sein et al. 2011, p.37).

Sein et al. (2011) criticise stage-gate models and suggest organising the building and evaluation of the IT artefact as two inseparable activities. We found that challenge 2 is mainly supported by the following principles: Authentic and Concurrent Evaluation, and Guided Emergence. We also experienced that the emphasis on integrating building and evaluation supported efficient and effective intervention. Integration also helped us to reflect upon the IT artefact and not only to build it. Apparently, evaluation is considered important in ADR and, thus, it is surprising that there is no proposed guidance for evaluation. As mentioned above, we used the strategies for evaluation suggested by Pries-Heje et al. (2008) and Venable (2016).

Another observation concerns the development of design principles. We chose to discuss this matter in relation to challenge 2 since the design principles emanated from the process of building and evaluating the IT artefact. In the ADR project, the design principles were formulated in parallel with the IT artefact for service assessment and innovation. In other words, the design principles emerged from the evaluation of the IT artefact and were gradually refined and re-inscribed into the IT artefact. In this way, there was a dialectic relationship between the development of the IT artefact and the development of the design principles. We appreciated the recommendation to develop design principles because it meant that we moved from the unique and specific to the abstract and generic. However, as mentioned in section 4.2.7, ADR does not provide prescriptive guidance regarding how to move to the abstract and generic, which led to our consultation of Gregor and Jones (2007) and Gregor and Hevner (2013).

As mentioned in section 5, there is also a lack of prescriptive guidance on how to formulate design principles. It seems that the ADR view on design principles is limited to including properties such as the form and function of the IT artefact. The design principles developed in the ADR project included both material properties and *process properties* (e.g., design principles concerning the collaboration aspect between service providers and customers). We were thus inspired by other perspectives on design principles which also acknowledge that design principles can include a method, process or activity (e.g., Walls et al. 1992; van den Akker 1999; van Aken 2004; Chandra et al. 2016; Cronholm and Göbel 2018). In the literature study (see section 2), we identified several publications that have used ADR to develop design principles of some kind. In addition, the literature study revealed that there are various ways of formulating design principles. This variation may obstruct the appropriate use of the design principles, due to a lack of consistency and/or omission of expected structure and content. Our empirical experiences and review of the literature indicate that a consistent formulation of design principles would increase their appropriate use. The shortcomings discussed above have prompted the following guidelines:

Guidelines concerning enhancements to the ADR method
In order to provide a generalisation of design principles, the ADR method should:

- Include prescriptive guidance on *how* to move from the specific and unique to the generic and abstract.

- Offer references to suggestions on how to formulate design principles (e.g., Walls et al. 1992; van den Akker 1999; van Aken 2004; Chandra et al. 2016; Cronholm and Göbel 2018).

Guidelines concerning how to apply ADR in practice

- In order to generalise contributions such as design principles, consult suggestions concerning abstraction mechanisms in terms of generalisation and specialisation (e.g., Gregor and Jones 2007; Gregor and Hevner 2013, Baskerville et al. 2018).

- In order to increase the utility of design principles, be consistent in their formulation (e.g., Walls et al. 1992; van den Akker 1999; Gregor and Jones 2007; Chandra et al. 2016; Cronholm and Göbel 2018).

6.3 Challenge 3: Intervention in the organisational context

"[W]e argue that a solution to this problem [the relevance challenge] requires a DR method that simultaneously aims at building innovative IT artifacts in an organizational context and learning from the intervention while addressing a problematic situation." (Sein et al. 2011, p.38)

Sein et al. (2011) criticise current DSR methods for not paying sufficient attention to intervention in organisations. The authors' response to challenge 3 is to integrate the AR concept of intervention into DSR. Our overall experience is that ADR was useful in solving problems that required organisational intervention. We have found that challenge 3 in one way or another is supported by all the ADR principles. Undoubtedly, the intervention strengthened the collaboration with the organisations, as well as supported learning and an in-depth understanding of the artefact-context relationship. As mentioned in section 2, Papas et al. (2012, p.156) concluded that ADR pays considerable attention to reflection and learning. We agree with this conclusion and we can add that ADR supports theoretical knowledge creation and maintains the balance between rigour and relevance.

In section 5.4, we described that a majority of the organisations were highly motivated in finding a general solution since this provided an opportunity to learn from each other. However, some of the organisations were primarily interested in a solution that addressed their own specific business problem. Unfortunately, ADR does not provide support regarding how to engage several organisations in the development of a general solution. In order to solve this problem, all the participating organisations were invited to workshops. The purpose of the workshops was to reflect upon the development and use of the IT artefact from all the organisations' perspectives. As mentioned above, the practitioners' practical knowledge was crucial for inscribing various forms of the organisational context into the IT artefact. The sharing of knowledge in the workshops meant that organisations learned from each other and not just from dyadic researcher-practitioner interventions. Additional learning from other organisations meant that the organisations and the ADR project as a whole gained a generic and abstract understanding of both the problem and the solution. Based on the discussion above, we propose the following guidelines:

Guidelines concerning enhancements to the ADR method

- In order to support ADR projects consisting of several client organisations, the ADR method should include advice and/or references concerning how to apply ADR in a context consisting of multiple client organisations.

Guidelines concerning how to apply ADR in practice
In order to support the development of a general solution when several organisations are involved:

- Establish arenas that include representatives from all the participating organisations as a complement to dyadic intervention between researchers and one organisation.

- Make use of knowledge concerning participatory action research (e.g., Whyte 1991) and participatory action design research (Bilandzic and Venable 2011).

7 Conclusion

The purpose of this study was to empirically evaluate ADR. Our literature review revealed that there is a lack of thorough systematic evaluations of ADR. In this study, we have presented empirical evidence, based on experiences from an ADR project. One overall conclusion concerning the usefulness of the ADR method is that it is highly relevant to an applied discipline such as IS, as IS researchers are expected to fulfil the dual mission of advancing theory while assisting practitioners in solving current and anticipated problems. In order to draw more specific conclusions with respect to ADR, we return to the three challenges formulated by Sein et al. (2011) justifying ADR (see section 6). We can conclude that ADR at a macro level:

(a) Supported the building and evaluation of an ensemble IT artefact (the IT artefact for service assessment and service innovation) and that the ensemble IT artefact was shaped by the organisational context and theoretical insights. Shaping relied both on the design and evaluation of the IT artefact, and also on influences from researchers and practitioners.

(b) Provided guidance for building, intervention, and evaluation in a collaborative effort. Activities in building the IT artefact, intervening in the organisations and evaluating the IT artefact were conducted concurrently. In other words, ADR overcame existing stage-gate models for DSR.

(c) Supported the adoption of an ontological position where organisational interventions and practitioner collaboration are emphasised. In this way, ADR ensured the relevance aspect and supported IS practitioners in solving immediate problems. As mentioned in section 1, Sein et al. (2011) state that ADR is needed because individually DSR and AR offer incomplete solutions. Based on the empirical evidence, our conclusion is that the integration of DSR and AR was fruitful.

However, support with respect to (a)-(c) above is provided at a macro level which needs to be operationalised and facilitated by complementing guidelines (see section 6). In section 5, we presented empirical evidence concerning the lack of prescriptive guidance in ADR. We acknowledge that the number of pages in journals is often limited, which compels authors to prioritise. Nevertheless, it is surprising that ADR does not sufficiently exploit existing knowledge with respect to operationalisation. We do not argue that ADR should include complete theories, methods or guidelines for overcoming the lack of prescriptive guidance. Instead, we suggest that literature references to related and valuable resources that provide prescriptive guidance should be included. In the ADR project, there was a specific need to consult other resources that recommended how to: 1) intervene in organisations in order to inscribe organisational factors into the IT artefact, 2) identify and use appropriate evaluation strategies, 3) use abstraction mechanisms in order to move from the specific and unique to the generic and abstract, and 4) formulate design principles. To meet this need, we propose guidelines that either complement the ADR method or provide guidance on how to apply ADR (see section 6).

We can conclude that none of the articles included in the literature review conducted a systematic evaluation of ADR's support for building and evaluating an ensemble artefact (see section 2). In addition, there is no prior study that offers prescriptive guidance regarding how ADR can actually be used. Therefore, in this respect, we posit that our study advances the state of the art. The conclusions are based on experiences gained from a single ADR project that included eight organisations and researchers. Due to this fact, we recommend future research using the results of this study as input for a survey study in which information systems researchers can evaluate agreement and disagreement on the basis of various projects. Finally, we hope that the findings are interesting enough to be considered in future ADR projects and ADR improvements.

References

Ågerfalk, P. J., (2019). Stimulating academic discourse: a call for response. *European Journal of Information Systems*, (28:1): 1-5.

Baskerville, R., Baiyere, A., Gregor, S., Hevner, A., and Rossi, M., (2018). Design science research contributions: finding a balance between artifact and theory. *Journal of the Association for Information Systems*, (19:5): 358-376.

Bilandzic, M., and Venable, J., (2011). Towards participatory action design research: adapting action research and design science research methods for urban informatics. *The Journal of Community Informatics*, (7:3). http://ci-journal.net/index.php/ciej/article/view/786. Retrieved July 8, 2017.

Chandra Kruse, L., Seidel, S., and Purao, S., (2016). Making use of Design Principles. In: *DESRIST2016*: *Proceedings of the 11th International Conference on Design Science Research in Information Technology and Systems*, J. Parsson, T. Tuunanen, J. Venable, B. Donellan, M. Helfert and J. Kenneally (eds.), Springer, pp. 37-51.

Cheng, X., Fu, S., Huang, J and de Vreede, G.J., (2018). Can Process Facilitation Improve Globally Distributed Collaboration? An Action Design Research. In: *HICSS51: Proceedings of the 51st Hawaii International Conference on System Sciences*, T. Bui (ed.), pp. 51.

Chesbrough, H., (2011). *Open Services Innovation: Rethinking Your Business to Grow and Compete in a New Era*, Jossey-Bass, San Francisco.

Collatto, D. C., Dresch, A., Lacerda, D. P., and Bentz, I. G., (2017). Is Action Design Research Indeed Necessary? Analysis and Synergies Between Action Research and Design Science Research. *Systemic Practice and Action Research*, (31:3): 239-267.

Conboy, K., (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information systems research*, (20:3): 329-354.

Cronholm, S., and Göbel, H., (2013). Action Design Research: Expanding the Scope. In: *ADWI: Proceedings of IT Artefact Design and Workpractice Intervention*, http://www.vits.org/?pageId=407&UltimateMenu1B=Item17. Retrieved July 2, 2019.

Cronholm S., and Göbel, H., (2016). ITIL Compliance with a Service Perspective: a Review Based on Service-Dominant Logic. In: *ACIS 2016: Proceedings of the 27th Australasian Conference on Information Systems*. https://ro.uow.edu.au/acis2016/papers/1/7/. Retrieved July 2, 2019.

Cronholm, S., and Göbel, H., (2018). Guidelines Supporting the Formulation of Design Principles. In: *ACIS 2018: Proceedings of 29th Australasian Conference on Information Systems. http://www.acis2018.org/wp-content/uploads/2018/11/ACIS2018_paper_24.pdf*. Retrieved July 2, 2019.

Cronholm S., Göbel, H., and Hjalmarsson, A., (2016). Evaluation of Action Design Research. In: *ACIS 2016: Proceedings of the 27th Australasian Conference on Information Systems*. https://ro.uow.edu.au/acis2016/papers/1/11/. Retrieved July 2, 2019.

Date, C. J., (2006). *An introduction to database systems*. Pearson Education, India.

DeGrace, P., and Stahl, L. H., (1990). *Wicked Problems, Righteous Solutions*, Prentice Hall, New Jersey.

DeGroot, M. H., (1974). Reaching a consensus. *Journal of the American Statistical Association*, (69:345): 118-121.

Ebel, P., Bretschneider, U., and Leimeister, J. M., (2016). Leveraging virtual business model innovation: a framework for designing business model development tools. *Information Systems Journal*, (26:5): 519-550.

Eisenhardt, K.M., and Graebner, M.E., (2007). Theory Building from Cases: Opportunities and Challenges. *Academy of Management Journal*, (50:1): 25-32.

Fitzgerald, B., Russo, N., Stolterman, E., (2002). Information Systems Development: Methods in Action. McGraw-Hill, London.

Giessmann, A., and Legner, C., (2016). Designing business models for cloud platforms. *Information Systems Journal*, (26:5): 551-579.

Göbel, H., and Cronholm, S., (2016). Nascent Design Principles Enabling Digital Service Platforms. In: *DESRIST2016: Proceedings of the 11th International Conference on Design Science Research in Information Technology and Systems,* J. Parsson, T. Tuunanen, J. Venable, B. Donellan, M. Helfert and J. Kenneally (eds.), Springer, pp. 52-67.

Goldkuhl, G., (1999). *The Grounding of Usable Knowledge: An inquiry in the epistemology of action knowledge*. CMTO Research Papers, 3. Linköping University, Sweden.

Goldkuhl, G., (2004). Design Theories in Information Systems—A Need for Multi-Grounding. *Journal of Information Technology Theory and Application (JITTA)*, (6:2): 59-72.

Gregor, S., and Hevner, A. R., (2013). Positioning and Presenting Design Science Research for Maximum Impact. *MIS quarterly*, (37:2): 337-355.

Gregor, S., Imran, A., and Turner, T., (2014). A 'sweet spot' change strategy for a least developed country: leveraging e-Government in Bangladesh. *European Journal of Information Systems*, (23:6): 655-671.

Gregor, S., and Jones, D., (2007). The anatomy of a design theory. *Journal of the Association for Information Systems*, (8:5): 312-335.

Haj-Bolouri, A., Bernhardsson, L., and Rossi, M., (2016). PADRE: A Method for Participatory Action Design Research. In: *DESRIST2016: Proceedings of the 11th International Conference on Design Science Research in Information Technology and Systems,* J. Parsson, T. Tuunanen, J. Venable, B. Donellan, M. Helfert and J. Kenneally (eds.), Springer, pp. 19-36.

Haj-Bolouri, A., Purao, S., Rossi, M., and Bernhardsson, L., (2017). Action Design Research as a Method-in-Use: Problems and Opportunities. In: *DESRIST2017: Proceedings of the 12th International Conference in Design Science Research in Information Technology and Systems*, A. Maedche, J. vom Brocke and A. Hevner (eds.), Springer, pp. 110-118

Haj-Bolouri, A., Purao, S., Rossi, M., and Bernhardsson, L., (2018). Action Design Research in Practice: Lessons and Concerns, In: *ECIS2018: Proceedings of the Twenty-Sixth European Conference on Information Systems,* P. M. Bednar, F. Ulrich and K. Kautz (eds.).

Hevner, A. R., (2007). A three cycle view of design science research. *Scandinavian Journal of Information systems*, (19:2): 87-92.

Hevner, A. March, S. T., Park, J., and Ram, S., (2004). Design Science in Information Systems Research. *MIS Quarterly*, (28:1): 75-105.

Hsieh, H. F., and Shannon, S. E., (2005). Three approaches to qualitative content analysis. *Qualitative Health Research*, (15:9): 1277-1288.

Huysmans, P. and De Bruyn, P., (2013). A Mixed Methods Approach To Combining Behavioral And Design Research Methods In Information Systems Research. In: *ECIS2013: Proceedings of the 21st European Conference on Information Systems,* p. 29.

Iivari, J., (2007). A Paradigmatic Analysis of Information Systems as a Design Science. *Scandinavian Journal of Information Systems,* (19:2): 39-63.

Iivari, J., (2015). Distinguishing and contrasting two strategies for design science research. *European Journal of Information Systems*, (24:1): 107-115.

Iivari, J., Maansaari. J., (1998). The usage of systems development methods: Are we stuck to old practices? *Information and Software Technology*, (40:9): 501-510.

ISO/IEC 20000-1 (2011), *Information Technology—Service Management—Part 1: Specification, and Part 2: Code of Practice,* International Standards Organization, Geneva, Switzerland.

Karu, K., Ferris, K., Hunnebeck, L., Rae, B. and Rance, S., (2016). *ITIL Practitioner*. The Stationery Office, London.

Keijzer-Broers, W., Florez-Atehortua, L., and de Reuver, M., (2016). Prototyping a Health and Wellbeing Platform: An Action Design Research approach. In: *HICSS49, Proceedings of the 49th Hawaii International Conference on System Sciences*, T. X. Bui and R. H. Sprague Jr.(eds.), pp. 3462-3471.

Lusch, R. F., and Nambisan, S., (2015). Service innovation: A service-dominant logic perspective. *MIS quarterly*, (39:1): 155-175.

March S. T., and Smith G., (1995). Design and Natural Science Research on Information Technologies. *Decision Support Systems,* (15:4): 251-266.

Mayring, P., (2000). Qualitative content analysis. *Forum: Qualitative Social Research*, *1*(2), http://www.qualitative-research.net/index.php/fqs/article/view/1089/2385. Retrieved July 8, 2017

Mettler, T., (2017) Contextualizing a professional social network for health care: Experiences from an action design research study. *Information Systems Journal,* (28:4): 684-707.

Mullarkey, M. T., and Hevner, A. R., (2015). Entering Action Design Research, In: *DESRIST2015: Proceedings of the tenth International Conference on Design Science Research in Information Systems and Technology*, B. Donellan, M. Helfert, J. Keneally, D. VanderMeer, M. Rothenberger and R. Winter (eds.), Springer, pp. 121-134.

Mullarkey, M. T., and Hevner, A. R., (2019). An elaborated action design research process model. *European Journal of Information Systems*, (28:1): 6-20.

Mustafa, M. I., and Sjöström, J., (2013). Design principles for research data export: lessons learned in e-health design research, In: DESRIST2013: *Proceedings of the eightth International Conference on Design Science Research in Information Systems and Technology,* J. vom Brocke, R. Hekkala, S. Ram and M. Rossi (eds.), Springer, pp 34-49.

Niemi, E., and Laine, S., (2016). Competence Management System Design Principles: Action Design Research, *In: ICIS2016: Proceedings of the International Conference of Information Systems*, P. J. Ågerfalk, N. Levina and S. S. Kien (eds.).

Orlikowski, W. J., and Iacono, C. S., (2001). Research Commentary: Desperately Seeking the 'IT' in IT Research—A Call to Theorizing the IT Artifact. *Information Systems Research,* (12:2): 121-134.

Papas, N., O'keefe, R. M., and Seltsikas, P., (2012). The action research vs design science debate: reflections from an intervention in eGovernment. *European Journal of Information Systems*, (21:2): 147-159.

Peffers, K., Tuunanen, T., and Niehaves, B., (2018). Design science research genres: introduction to the special issue on exemplars and criteria for applicable design science research, *European Journal of Information Systems*, (27:2), pp. 129-139

Peffers, K., Tuunanen, T., Rothenberger, M. A., and Chatterjee, S., (2007). A design science research methodology for information systems research. *Journal of Management Information Systems*, (24:3): 45-77.

Pries-Heje, J., Baskerville, R., Venable, J.R.: Strategies for Design Science Research Evaluation. *In: ECIS 2008: Proceedings of the 16th European Conference on Information Systems*, W. Golden, H. van der Heijden and V. K. Tuunainen (eds.).

Rogerson, C., and Scott, E., (2014). Motivating an action design research approach to implementing online training in an organisational context. *Interactive Technology and Smart Education,* (11:1): 32-44.

Schuppan, T., and Koehl, S., (2017). One Stop Government: Stalled Vision or a Matter of Design?—Empirical Findings from Social Services in Germany. In: *HICSS50: Proceedings of the 50th Annual Hawaii International Conference on System Sciences.* T. Bui (ed.).

Sein, M. K., Henfridsson, O., Purao, S., Rossi, M., and Lindgren, R., (2011). Action Design Research. *MIS Quarterly*. (35:1): 37-56.

Sein, M. K., and Rossi, M., (2019). Elaborating ADR while drifting away from its essence: A commentary on Mullarkey and Hevner. *European Journal of Information Systems*, (28:1): 21-25.

Seuring, S. and Müller, M., (2008), From a literature review to a conceptual framework for sustainable supply chain management. *Journal of cleaner production*, (16:15): 1699-1710.

Spagnoletti, P., Resca, A., and Sæbø, Ø., (2015). Design for social media engagement: insights from elderly care assistance. *The Journal of Strategic Information Systems*, (24:2): 128-145.

Sun, Y., and Kantor, P. B., (2006). Cross-Evaluation: A new model for information system evaluation. *Journal of the Association for Information Science and Technology*, (57:5): 614-628.

Team, C. P., (2010). *CMMI for Service, Version 1.3, CMMI-SVC v1. 3*. CMU/SEI-2010-TR-034, Technical report, Software Engineering Institute.

Van Aken, J. 2004, Management Research Based on the Paradigm of the Design Sciences: The Quest for Field-tested and Grounded Technological Rules, *Journal of Management Studies*, (41:2), 219-246.

Van den Akker, J. 1999. Principles and methods of development research. *Design approaches and tools in education and training*. Springer Science and Business Media, pp 1-14.

Vargo, S.L., and Lusch, R.F., (2008a) Why Service?, *Journal of the Academy of Marketing Science*, (36:1): 25-38.

Vargo, S. L., and Lusch, R. F., (2008b). Service-dominant logic: continuing the evolution. *Journal of the Academy of Marketing Science,* (36:1): 1-10.

Veling, L., Mc Quillan, L., Browne, A., Craig, M., and Pinkster, M., (2016). Use It or Lose It: Embodying Practice in Action Design Research (ADR), *In: ICIS2016: Proceedings of the International Conference of Information Systems*, P. J. Ågerfalk, N. Levina and S. S. Kien (eds.).

Venable, J., Pries-Heje, J. and Baskerville, R., (2016). FEDS: a framework for evaluation in design science research, *European Journal of Information Systems,* (25:1): 77-89.

Vom Brocke, J., and Rosemann, M., (2010). *Handbook on business process management*, Springer, Heidelberg.

Walls, J. G., Widmeyer, G. R., and El Sawy, O. A., (1992). Building an information system design theory for vigilant EIS. *Information systems research*, (3:1): 36-59.

Walsham, G., (1995). Interpretive case studies in IS research: Nature and method. *European Journal of Information Systems*, (4:2): 74-81.

Westin, S., and Sein, M.K., (2015). The Design and Emergence of a Data/Information Quality System, *Scandinavian Journal of Information Systems*, (27:1), 3-26.

Whyte, W. F. E., (1991). *Participatory Action Research*. Sage Publications, Inc.

William, D., and Black, P., (1996). Meanings and consequences: a basis for distinguishing formative and summative functions of assessment? *British Educational Research Journal* (22:5): 537-548.

Wilson, P. F., Dell, L. D., and Anderson, G. F., (1993). Root Cause Analysis: A Tool for Total Quality Management. ASQ Quality Press.

Zimmermann, A., Lorenz, A., and Oppermann, R., (2007). An operational definition of context. In *International and Interdisciplinary Conference on Modeling and Using Context,* B. Kokinov, D. C. Richardson, T. R. Roth-Berghofer and L. Vieu (eds.), Springer, pp. 558-571.